CLAIMS

Sub B7

1. A method of transmitting information between a first computer and a second computer, comprising the steps of:

(1) embedding in each of a plurality of data packets a discriminator value that periodically changes between successive data packets, wherein each discriminator value is not based solely on the value of other data in each data packet;

(2) transmitting the plurality of data packets between the first computer and the second computer;

(3) receiving the transmitted data packets at the second computer; and

(4) for each received data packet, comparing the discriminator value to a set of valid discriminator values and, in response to detecting a match, accepting the received data packet for further processing, and otherwise rejecting the received data packet.

2. The method of claim 1, wherein step (1) comprises the step of using an Internet Protocol address in an Internet Protocol header as the discriminator value, wherein the Internet Protocol address is used to route the data packets over the Internet.

3. The method of claim 2, further comprising the step of changing in value only part of the Internet Protocol addresses between successive packets.

Sub B7

4. The method of claim 1, further comprising the step of using as the discriminator value a data field external to an Internet Protocol header of each data packet.

5. The method of claim 1, wherein steps (1) and (4) are performed in a data link layer of an ISO standard communication protocol.

6. The method of claim 1, wherein step (1) comprises the step of using a Media Access Control (MAC) hardware address as the discriminator value, wherein the MAC hardware address is used to route the data packets on a local area network.

7. The method of claim 1, wherein step (1) comprises the step of using a different discriminator value for each successive packet.

8. The method of claim 1, wherein step (4) comprises the step of comparing each

discriminator value to a window of valid discriminator values, wherein the window is wide enough to permit comparison to only a small number of potentially valid discriminator values, and further comprising the step of moving the window as successive data packets are received.

9. The method of claim 1, further comprising the step of sharing between the first computer and the second computer information sufficient to generate the set of valid discriminator values.

10. The method of claim 1, further comprising the step of transmitting from the first computer to the second computer an algorithm for selecting successively valid discriminator values.

11. The method of claim 1, wherein step (4) comprises the step of using a presence vector to determine whether to accept each data packet.

12. The method of claim 1, wherein step (4) comprises the step of using a hashing function to determine whether the discriminator value is valid.

13. The method of claim 1, further comprising the step of transmitting a synchronization request between the first computer and the second computer, wherein the second computer uses the synchronization request to maintain synchronization of valid discriminator values.

14. The method of claim 13, further comprising the step of, in response to failure to receive a synchronization acknowledgement from the second computer, shutting off transmission of data packets to the second computer.

15. The method of claim 13, further comprising the step of embedding a synchronization value in each data packet that permits the second computer to re-establish synchronization in a set of potentially valid discriminator values.

16. The method of claim 13, further comprising the step of moving a window of valid discriminator values in the second computer in response to receiving the synchronization request from the first computer.

17. The method of claim 1, wherein step (1) comprises the steps of using an Internet Protocol source address in an Internet Protocol header as a first part of the discriminator value

and using an Internet Protocol destination address in the Internet Protocol header as a second part of the discriminator value, wherein the source and destination addresses are used to route each data packet over the Internet.

18. The method of claim 17, further comprising the steps of:

embedding a plurality of the data packets into a frame; and

embedding a source and destination hardware address in the frame, wherein the source and destination hardware address are quasi-randomly generated and used to route the frame on a network.

19. The method of claim 1, further comprising the step of maintaining in the first computer a first transmit table ~~and a first~~ receive table, and maintaining in the second computer a second transmit table and a second receive table,

wherein each transmit table comprises a list of valid discriminator values that are to be inserted into outgoing data packets;

wherein each receive table comprises a list of valid discriminator values that are to be compared against incoming data packets; and

wherein the first transmit table in the first computer matches the second receive table in the second computer; and wherein the first receive table in the first computer matches the second transmit table in the second computer.

20. A method of transmitting data packets over a network comprising a plurality of computers connected to each other through a plurality of physical transmission paths, the method comprising the steps of:

(1) for each of the plurality of data packets, randomly selecting one of the plurality of physical transmissions paths through the plurality of computers; and

(2) transmitting each data packet over the randomly selected physical transmission path.

21. The method of claim 20, wherein step (1) comprises the steps of:

(a) selecting a path defined by a pair of computers in the network;

(b) selecting valid source and destination addresses associated with the selected path; and

(c) inserting the valid source and destination addresses into the data packet before transmitting it over the selected path.

22. The method of claim 21, wherein step (1) comprises the step of avoiding selection of a path that is not operational.

23. A system comprising:

a first computer that embeds into each of a plurality of data packets a discriminator value that periodically changes between successive data packets, wherein each discriminator value is not based solely on the value of other data in each data packet; and

B2

a second computer coupled to the first computer through a network,

wherein the first computer transmits the plurality of data packets to the second computer, and

wherein the second computer receives the transmitted data packets, compares the discriminator value in each received data packet to a set of valid discriminator values and, in response to detecting a match, accepts the received data packet for further processing, and otherwise rejects the received data packet.

24. The system of claim 23, wherein the first computer embeds into each of the plurality of data packets an Internet Protocol address in an Internet Protocol header as the discriminator value, wherein the Internet Protocol address is used to route the data packets over the Internet.

25. The system of claim 24, wherein the first computer changes in value only part of the Internet Protocol addresses between successive packets.

26. The system of claim 23, wherein the first computer embeds the discriminator value in a data field external to an Internet Protocol header of each data packet.

B2

27. The system of claim 23, wherein the first computer embeds each discriminator value in a first data link layer of an ISO standard communication protocol, and wherein the second computer compares each discriminator value in a second data link layer of the ISO standard communications protocol.

28. The system of claim 23, wherein the first computer embeds a Media Access Control (MAC) hardware address as the discriminator value, wherein the MAC hardware address is used to route the data packets on a local area network.

29. The system of claim 23, wherein the first computer embeds a different discriminator value for each successive packet.

30. The system of claim 23, wherein the second computer compares each discriminator value to a window of valid discriminator values, wherein the window is wide enough to permit comparison to only a small number of potentially valid discriminator values, and wherein the window is moved as successive data packets are received.

B2

31. The system of claim 23, wherein the first and second computers share common information sufficient to generate the set of valid discriminator values.

32. The system of claim 23, wherein the first computer transmits to the second computer an algorithm for selecting successively valid discriminator values.

33. The system of claim 23, wherein the second computer uses a presence vector to determine whether to accept each data packet.

34. The system of claim 23, wherein the second computer uses a hashing function to determine whether the discriminator value is valid.

35. The system of claim 23, wherein the first computer transmits to the second computer a synchronization request, wherein the second computer uses the synchronization request to maintain synchronization of valid discriminator values.

36. The system of claim 35, wherein the first computer, in response to failure to receive a synchronization acknowledgement from the second computer, shuts off transmission of data packets to the second computer.

37. The system of claim 35, wherein the first computer embeds a synchronization value in each data packet that permits the second computer to re-establish synchronization in a set of potentially valid discriminator values.

38. The system of claim 35, wherein the second computer moves a window of valid discriminator values in response to receiving the synchronization request from the first computer.

39. The system of claim 23, wherein the first computer embeds an Internet Protocol source address in an Internet Protocol header as a first part of the discriminator value and embeds an Internet Protocol destination address in the Internet Protocol header as a second part of the discriminator value, wherein the source and destination addresses are used to route each data packet over the Internet.

40. The system of claim 39, wherein the first computer embeds a plurality of the data packets into a frame and embeds a source and destination hardware address in the frame, wherein the source and destination hardware address are quasi-randomly generated and used to route the frame on a network.

B3

41. The system of claim 23,

wherein the first computer comprises a first transmit table and a first receive table,

wherein the second computer comprises a second transmit table and a second receive table,

wherein each transmit table comprises a list of valid discriminator values that are to be inserted into outgoing data packets,

wherein each receive table comprises a list of valid discriminator values that are to be compared against incoming data packets,

wherein the first transmit table in the first computer matches the second receive table in the second computer, and

wherein the first receive table in the first computer matches the second transmit table in the second computer.

42. A first computer coupled to a network comprising a plurality of computers connected to each other through a plurality of physical transmission paths,

wherein the first computer generates a plurality of data packets for transmission across the network; and wherein the first computer, for each of the plurality of data packets, randomly selects one of the plurality of physical transmissions paths through the plurality of computers and transmits each data packet over the randomly selected physical transmission path.

43. The first computer of claim 42, wherein the first computer:

(a) selects a path defined by a pair of computers in the network;

(b) selects valid source and destination addresses associated with the selected path; and

(c) inserts the valid source and destination addresses into the data packet before transmitting it over the selected path.

Sub A1

44. The first of claim 43, wherein the first computer avoids the step of avoiding selection of a path that is not operational.

A

45. A system comprising in combination:

B2

a transmitting node that generates pseudo-random discriminator values and embeds the pseudo-random discriminator values into data packets for transmission; and

a receiving node that receives data packets transmitted by the transmitting node, wherein the receiving node, for each received packet, extracts the pseudo-randomly generated discriminator value, compares it to a set of potentially valid discriminator values shared between the transmitting node and the receiving node and, in response to detecting a match, accepts the data packet, and otherwise discards the packet.

46. The system of claim 45, wherein the receiving node maintains a window of valid discriminator values, wherein the window is moved in response to detecting a match.

47. The system of claim 45, wherein each pseudo-randomly generated discriminator value comprises a valid Internet Protocol address that is assigned to the receiving node.

48. The system of claim 45, wherein each pseudo-randomly generated discriminator value comprises a valid Media Access Control (MAC) hardware address that is assigned to the receiving node.

49. The system of claim 45, wherein the transmitting node generates a different pseudo-randomly generated discriminator value for each successive data packet.

50. A receiving computer that receives data packets from a transmitting computer, wherein the receiving computer comprises computer instructions that execute the steps of:

(1) for each received data packet, extracting a discriminator value inserted by the transmitting computer;

(2) comparing the extracted discriminator value to a set of valid discriminator values on the basis of information previously shared with the transmitting computer; and

(3) in response to detecting a match in step (2), accepting the received data packet for further processing and otherwise rejecting the data packet.

51. The receiving computer of claim 50, wherein the receiving computer further comprises computer instructions that extract as the discriminator value an Internet Protocol address from a header portion of each data packet.

52. The receiving computer of claim 50, wherein the receiving computer maintains a window of valid discriminator values, wherein the window is moved in response to detecting matches.

53. The receiving computer of claim 50, wherein the receiving computer receives information from the transmitting computer sufficient to establish the set of valid discriminator values.

54. A method of transmitting data from a first computer to a second computer, the data comprising a plurality of data bytes arranged in a particular order, the method comprising the steps of:

(1) establishing in the first computer and second computer a common algorithm that determines how data will be randomly distributed across a plurality of data packets;

(2) in the first computer, randomly distributing the plurality of data bytes across the plurality of data packets according to the common algorithm;

(3) transmitting the plurality of data packets from the first computer to the second computer; and

(4) in the second computer, extracting the randomly distributed plurality of data bytes

B3

from the plurality of data packets and reassembling them into the particular order according to the common algorithm.

Sub A2

55. The method of claim 1, wherein step (3) comprises the step of transmitting each of the plurality of data packets across a different path in a computer network.

56. A system comprising:

a first computer including an algorithm that establishes a random distribution pattern for allocating data across a plurality of data packets, wherein the first computer randomly distributes data bytes from a data source across the plurality of data packets according to the random distribution pattern and transmits the plurality of data packets across a network; and

a second computer coupled to the first computer across the network, wherein the second computer receives the plurality of data packets from the first computer, extracts the randomly distributed data bytes, and reassembles them into their original order according to the algorithm.

57. The system of claim 56, wherein the first computer transmits each of the plurality of data packets across a different path in the network.

58. A method of securely transmitting a data packet between a sending computer and a receiving computer, comprising the steps of:

(1) encrypting the data packet using a session key known to the sending computer and the receiving computer, but not known by intermediate computers between the sending computer and the receiving computer;

(2) adding a packet header that identifies the data packet to the data packet encrypted in step (1);

(3) encrypting the combined packet header and encrypted data packet created in step (2) using a link key known to each of a plurality of intermediate computers arranged between the first computer and the second computer;

(4) adding a cleartext packet header to route the packet encrypted in step (3); and

(5) transmitting the packet created in step (4).

59. The method of claim 58, further comprising the steps of:

OK for 7/31/18

12/8/02

(5) at each intermediate computer, decrypting the packet received from a previous computer and decrypting it using the link key;

(6) re-encrypting the packet using a different link key known to a next intermediate computer in the network;

(7) adding a cleartext packet header to route the packet re-encrypted in step (6); and

(8) transmitting the packet created in step (7) to the next intermediate computer.

60. The method of claim 59, further comprising the step of, at the receiving computer, decrypting the packet using the session key.

61. A method of transmitting data over a computer network, comprising the steps of:

at an originating terminal connected to the computer network, receiving a stream of data and forming first level data packet payloads therefrom;

identifying a network destination address for the stream of data and adding first level headers containing data representing the network destination address to each of the data packets to form a first level packet;

encrypting each of the first level packets to form second level packet payloads; attaching to the second level packet, payloads headers containing as destination addresses, addresses of at least one intermediate router connecting the originating terminal to the destination to form second level packets;

sending the second level packets to the at least one intermediate router;

at the at least one intermediate router, decrypting at least one of the second level payloads and determining from the first level headers the destination address, forming new packets containing at least the first level packet payloads, and attaching headers thereto containing the destination address, whereby a true destination of the data stream is concealed behind a layer of encryption for at least a portion of its travel over the network.

62. The method of claim 61, wherein the step of attaching includes determining the at least one intermediate router by randomly selecting from a group of intermediate routers.

63. The method of claim 61, wherein the step of determining from the first level headers

the destination address includes converting the data representing the network destination address with the network destination address by means of correlation data stored on the intermediate router.

64. The method of claim 61, further comprising the step of including in one of the first and second layer headers, an indicator of a number of hops to be made by the first level packet before arriving at the network destination, the at least one intermediate router decrementing the indicator of a number of hops and sending the first level packet to another intermediate router responsively to a value of the indicator of a number of hops.

65. A method of routing packets on a packet network, comprising the steps of: block-encrypting, with a session key, message data to form payloads;

dividing an encrypted block resulting from the block-encrypting into at least two data payloads such that interleaving portions of data resulting from the block-encrypting step are among the at least two data payloads;

encrypting, with a link key, each of the ~~at least~~ two data payloads, together with destination data identifying a final destination for the packets;

combining, with a first payload resulting from the last step of encrypting, a first hop address indicating a first intermediate destination address and transmitting a first packet resulting thereby to the first intermediate destination address;

combining, with a second payload resulting from the last step of encrypting, a second hop address indicating a second intermediate destination address and transmitting a second packet resulting thereby to the second intermediate destination address.

66. The method of claim 65, further comprising the steps of:

combining, in the first packet, a first hop counter;

at a terminal coinciding with the first intermediate destination address, determining, responsively to the first hop counter, to send the first packet to the final destination address; and

at the terminal coinciding with the first intermediate destination address, decrypting with the link key the first payload to expose the final destination address and sending the first packet to the final destination address, responsively to the step of determining.

67. The method of claim 65, further comprising the steps of:

combining, in the second packet, a second hop counter;

at a terminal coinciding with the second intermediate destination address, determining, responsively to the second hop counter, to send the first packet to the final destination address;

at the terminal coinciding with the second intermediate destination address, decrypting with the link key the second payload to expose the final destination address and sending the second packet to the final destination address, responsively to the last step of determining.

Add A3

Add B7